Global Texture Enhancement for Fake Face Detection In the Wild

Zhengzhe Liu, Xiaojuan Qi[1,2], Philip H. S. Torr[1]
[1]University of Oxford, [2]The University of Hong Kong

Abstract

Generative Adversarial Networks (GANs) can generate realistic fake face images that can easily fool human beings. On the contrary, a common Convolutional Neural Network (CNN) discriminator can achieve more than 99.9% accuracy in discerning fake/real images. In this paper, we conduct an empirical study on fake/real faces, and have two important observations: firstly, the texture of fake faces is substantially different from real ones; secondly, global texture statistics are more robust to image editing and transferable to fake faces from different GANs and datasets. Motivated by the above observations, we propose a new architecture coined as Gram-Net, which leverages global image texture representations for robust fake image detection. Experimental results on several datasets demonstrate that our Gram-Net outperforms existing approaches. Especially, our Gram-Net is more robust to image editings, e.g. down-sampling, JPEG compression, blur, and noise. More importantly, our Gram-Net generalizes significantly better in detecting fake faces from GAN models not seen in the training phase and can perform decently in detecting fake natural images.

1. Introduction

With the development of GANs [9, 12, 13, 1], computers can generate vivid face images that can easily deceive human beings as shown in Figure 1. (Can you guess which images are generated from GANs?) These generated fake faces will inevitably bring serious social risks, *e.g.* fake news and evidence, and pose threats to security. Thus, powerful techniques to detect these fake faces are highly desirable. However, in contrast to the intensive studies in GANs, our understanding of generated faces is fairly superficial and how to detect fake faces is still an under-explored problem. Moreover, fake faces in practical scenarios are from different unknown sources, *i.e.* different GANs, and may undergo unknown image distortions such as downsampling, blur, noise and JPEG compression, which makes this task even more challenging. In this paper, we aim to produce new insights on understanding fake faces from GANs and propose a new architecture to tackle the above challenges. Our contributions are as follows.

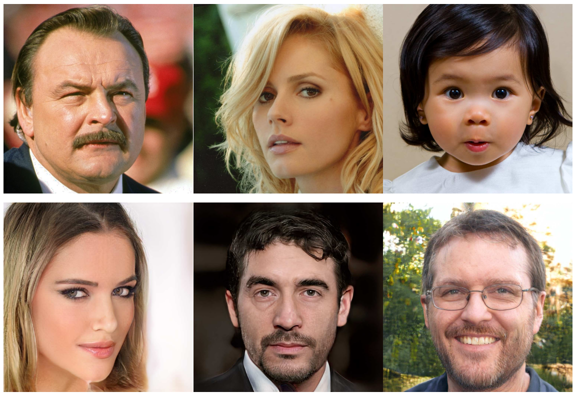


Figure 1. Can you determine which are real and which are fake? (answer key below)[1]

Contribution 1. To facilitate the understanding of face images from GANs, we systematically study the behavior of human beings and CNN models in discriminating fake/real faces detailed in Section 3.1. In addition, we conduct extensive ablation experiments to diagnose the CNN discriminator and perform low-level statistics analysis as verification.

These empirical studies lead us to the following findings.

- *Texture statistics* of fake faces are substantially different from natural faces.
- Human focus on visible shape/color artifacts to detect fake face while CNNs focus more on *texture* regions.
- CNNs take *textures* as an important cue for fake face detection. A ResNet model performs almost perfectly in detecting untouched fake faces if the training data and testing data are from the same source.

Contribution 2. Although a CNN based fake face detector performs significantly better than human beings, it is still not robust enough to handle real-world scenarios, where images may be modified and/or from different unknown sources. With further analysis of the relationship between *texture* and fake face detection, we found *large texture* information is

[1] The first three are real and the last three are fake.

more robust to image distortions and more invariant for face images from different GANs. However, CNNs cannot fully capture *long-range or global* cues due to their limited effective receptive field as studied in [21]. Motivated by the above observation, we further develop a novel architecture – Gram-Net, which improves the robustness and generalization ability of CNNs in detecting fake faces. The model incorporates "Gram Block" into the CNN backbone shown in Figure 5. The introduced Gram layer computes *global texture* representations in multiple semantic levels, which complements the backbone CNN.

Contribution 3 Experiments on fake faces from StyleGAN [13], PGGAN [12], DRAGAN [15], DCGAN [29], StarGAN [4], and real faces from CelebA-HQ [12], FFHQ [13], CelebA [20], show that our Gram-Net achieves state-of-the-art performance on fake face detection. Specifically, our proposed Gram-Net is robust for detecting fake faces which are edited by resizing (10% improvement), blurring (15% improvement), adding noise (13% improvement) and JPEG compressing (9% improvement). More importantly, Gram-Net demonstrates significantly better generalization abilities. It surpasses the compared approaches by a large margin (more than 10% improvement) to detect fake faces generated by GANs that are not seen in the training phase and GANs trained for other tasks including image-to-image translation GANs, *e.g*. StarGAN. Further, our experiments show that Gram-Net (trained on StyleGAN) generalizes much better with a 10% improvement to detect fake natural images from GANs trained on ImageNet [16], *e.g*. BigGAN [3].

2. Related work

GANs for human face generation. Recently, GAN models [8, 29, 15, 1, 2, 12, 13, 19, 36, 4] have been actively studied with applications for face image generation. One stream of research is to design GANs [8, 29, 15, 1, 2] for generating random face images from random vectors. Early works [8, 29, 15, 1, 2] can generate high quality low resolution images but suffer from mode collapse issues for generating high resolution images. The most advanced high resolution (1024 × 1024) GAN models – PGGAN [12] and StyleGAN [13]– can generate high quality face images that can even fool human beings. Another stream is to utilize GAN models for image-to-image translation tasks [19, 36, 4], *e.g*., Choi *et al*. proposed StarGAN model which can perform face image to face image translation. These generated fake faces may cause negative social impact. In this work, we aim to help the community gain more understanding about GAN generated fake faces and introduce novel neural network architecture for robust fake face image detection.

Fake GAN face detection. Recently, some researchers have investigated the problem of fake face detection [17, 26, 27, 23, 24, 32, 34, 30]. Color information is exploited in [17, 26]. In contrast, we found the performance of the CNN models changes little even if color information is removed. Marra *et al*. [23] showed that each GAN leaves specific finger-prints on images, and proposed to identify the source generating these images. However, the method cannot generalize to detect fake faces from GAN models that do not exist in the training data. Xuan *et al*. [32] adopted data augmentation for improving generalization, nevertheless, further improvements are limited by the detection algorithm. Nataraj *et al*. [27] proposed to take a color co-occurrence matrix as input for fake face detection. However, the hand-craft feature input results in losing the information of raw data. Zhang *et al*. [34] designed a model to capture the artifacts caused by the decoder. However, it failed to detect fake images from GANs with drastically different decoder architecture which is not seen in the training phase, while our approach can handle this case effectively. Wang *et al*. [30] proposed a neuron coverage based fake detector. However, the algorithm is time-consuming, hard to be deployed in real systems, and the performance is still far from satisfactory. Marra *et al*. [25] detected fake images with incremental learning. However, it only works when many GAN models are accessible in the training phase. Other works [18, 33] focused on the alignment of face landmarks to check whether the face is edited by face-swapping tools like DeepFakes [19]. Unlike the above, we intensively analyze fake faces, and correspondingly propose a novel simple framework which is more robust and exhibits significantly better generalization abilities.

Textures in CNNs. The texture response of CNNs has attracted increasing attention in the last few years. Geirho *et al*. [7] showed that CNN models are strongly biased on textures rather than shapes. Our empirical study also reveals that CNN can utilize texture for fake face detection which is in line with the findings in [7]. Motivated by the above observation, we further analyzed texture differences in terms of low-level statistics. Gatys *et al*. [5] proposed that the Gram matrix is a good description of texture, which is further utilized for texture synthesis and image style transfer [6]. The above works exploit the Gram matrix for generating new images by constructing Gram matrix based matching losses. Our work is related to these methods by resorting to the Gram matrix. However, different from [6, 5], our work adopts the Gram matrix as a global texture descriptor to improve discriminative models and demonstrates its effectiveness in improving robustness and generalization.

3. Empirical Studies and Analysis

3.1. Human vs. CNN

To shed insights on understanding fake faces generated form GANs, we systematically analyze the behavior of human beings and CNNs in discerning fake/real faces by conducting psychophysical experiments. Specifically, our experiments are performed in *in-domain* setting, where the model is trained and tested on fake images from the same GAN.

User study. For each participant, we firstly show him/her all the fake/real faces in the training set (10K real and 10K fake images). Then a randomly picked face image in our test set is shown to him/her without a time limit. Finally, he/she is required to click the "real" or "fake" button. On average, it takes around 5.14 seconds to evaluate one image. The results in this paper are based on a total of 20 participants, and each participant is required to rate 1000 images. At the same time, we also collected the user's judgment basis if his/her selection was "fake". According to their votings, human users typically take as evidence easily recognized shape and color artifacts such as "asymmetrical eyes", "irregular teeth", "irregular letters", to name a few.

CNN study and results. Testing images are also evaluated by CNN model – ResNet-18 [11]. The training and testing follow the *in-domain* setup. Table 1 (row1 & row2) shows that human beings are easily fooled by fake faces. In contrast, the ResNet CNN model achieves more than 99.9% accuracy in all experiments.

Analysis. To gain a deeper understanding about the question "Why CNNs perform so well at fake/real face discrimination?" and "What's the intrinsic difference between fake and real faces?", we further exploited CAM [35] to reveal the regions that CNNs utilize as evidence for fake face detection. Representative classification activation maps are shown in Figure 2. We can easily observe that the discriminative regions (warm color regions in Figure 2) for CNNs mainly lie in the *texture* regions, *e.g.* skin and hair, while the regions with clear artifacts make little contribution (cold color, red bounding box in Figure 2). The above observation motivates us to further study whether *texture* is an important cue that CNNs utilize for fake face detection and whether fake faces are different from real ones regarding *texture* statistics.

3.2. Is texture an important cue utilized by CNNs for fake face detection?

To validate the importance of textures for fake face detection, we conduct *in-domain* experiments on the skin regions since they contain rich texture information and less structural information such as shape. More specifically, we design the following controlled experiments on skin regions.

- *Original (skin)*: The input is the left cheek skin region based on DLib [14] face alignment algorithm as shown in Figure 3 (a – b). This is to verify whether the skin region contains enough useful information for fake face detection.
- *Gray-scale (skin)*: The skin regions are converted to gray-scale images. Typical examples are shown in Figure 3 (c – d). This experiment is to ablate the influence of color.
- *L0-filtered (skin)*: Small textures of the skin regions are filtered with L_0 filter [31].The L_0 algorithm can keep shape and color information while smoothing small textures. Typical examples are shown in Figure 3 (e – f).

Experimental results are shown in Table 1 (row 3 – row 5). The results of full image, original skin region, gray-scale skin region as inputs all indicate that skin regions already contain enough information for *in-domain* fake face detection and that colors do not influence the result much. The significant drop of performance (around 20%) of L_0 filtered inputs demonstrates the importance of texture for fake face detection in CNN models. In summary, texture plays a crucial role in CNN fake face detection and CNNs successfully capture the texture differences for discrimination, since the skin region performs on par with the full image in Table 1 (row 2 & row 3).

3.3. What are the differences between real & fake faces in terms of texture?

Empirical findings in Sec. 3.2 further motivate us to investigate the differences between real/fake faces in terms of texture. In the following, we adopt a texture analysis tool – the gray-level co-occurrence matrix (GLCM) [10].

The GLCM $P_\theta^d \in R^{256\times256}$ is created from a gray-scale texture image, and measures the co-occurrence of pixel values at a given offset parameterized by distance d and angle θ. For example, $P_\theta^d(i,j)$ indicates how often a pixel with value i and a pixel at offset (d,θ) with pixel value j co-exist. In our analysis, we calculate P_d^θ across the whole dataset to get the statistical results, where $d \in \{1,2,5,10,15,20\}$ and $\theta \in \{0,\pi/2,\pi,3\pi/2\}$ represents {right, down, left, upper}, d and θ can capture the property of textures with different size and orientation respectively. From the GLCM, we compute the texture contrast $\mathcal{C}_d$ at different distance offsets as follows,

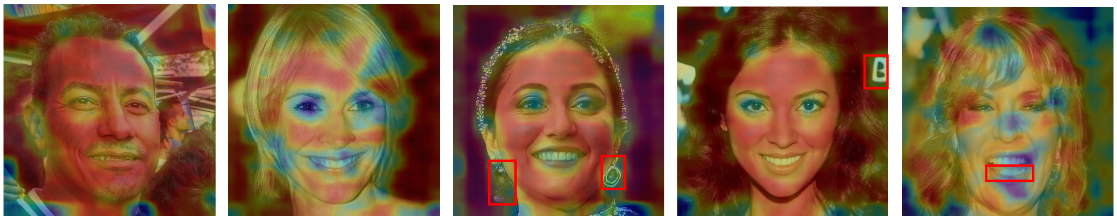
$$\mathcal{C}_d = \frac{1}{N}\sum_{i,j=0}^{255}\sum_{\theta=0}^{3\pi/2}|i-j|^2 P_d^\theta(i,j) \tag{1}$$

where $N = 256 \times 256 \times 4$ is a normalization factor, i, j represents pixel intensities, and d indicates pixel distances which are adopted to compute $\mathcal{C}_d$. Larger $\mathcal{C}_d$ reflects stronger texture contrast, sharper and clearer visual effects. Inversely, low value $\mathcal{C}_d$ means the texture is blurred and unclear.

The contrast component of GLCM is shown in Table 2. Real faces retain *stronger contrast* than fake faces at all measured distances. One explanation for this phenomenon is that

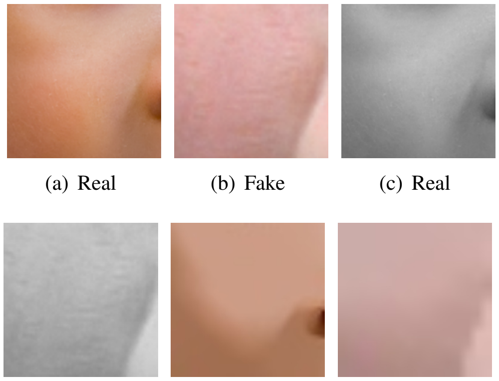
Input	Human vs. CNNs	StyleGAN vs. CelebA-HQ	StyleGAN vs. FFHQ	PGGAN vs. CelebA-HQ
Full image	Human Beings	75.15%	63.90%	79.13%
Full image	ResNet	99.99%	99.96%	99.99%
Original (skin)	ResNet	99.93%	99.61%	99.96%
Gray-scale (skin)	ResNet	99.76%	99.47%	99.94%
L0-filtered (skin)	ResNet	78.64%	76.84%	72.02%

Table 1. Quantitative results on fake face detection of human beings and CNNs, and skin region ablation studies in the *in-domain* setting.



(a) Real (b) Real (c) Fake (d) Fake (e) Fake

Figure 2. Class activation maps from trained ResNet model (better viewed in color). The red bounding box shows the visible artifacts indicated by human observers but activated weakly by CNN: (c) asymmetrical earrings; (d) irregular letter; (e) irregular teeth.



(a) Real (b) Fake (c) Real

(d) Fake (e) Real (f) Fake

Figure 3. Example images of Original (Skin) (a–b), Gray-scale (Skin) (c–d) and L0 filtered (Skin) (e–f).(better viewed in color)

the CNN based generator typically correlates the values of nearby pixels and cannot generate as strong texture contrast as real data. In this section, we only provide an analysis of texture contrast and admit that the differences between real and fake faces are definitely beyond our analysis. We hope this can stimulate future research in analyzing the texture differences for fake face detection.

4. Improved Model: Better Generalization Ability, More Robust

Until now, our analysis has been performed in the *in-domain* setting. The next step is to investigate the *cross-GAN* setting, where training and testing images are from different GAN models. Besides, we also investigate the images which are further modified by unintentional changes such as downsampling, JPEG compression and/or even intentional editing by adding blur or noise. Our following analysis remains to focus on *texture* due to our findings in Sec. 3.1 – Sec. 3.3.

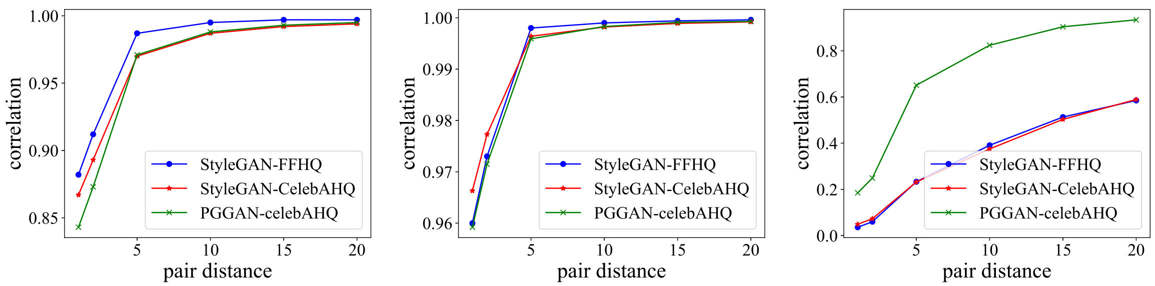
Dataset \ distance (d)	1	2	5	10	15	20
CelebA-HQ	**8.68**	**12.37**	**61.52**	**117.94**	**181.30**	**237.30**
StyleGAN(on CelebA-HQ)	4.92	8.84	47.40	93.79	146.33	193.49
PGGAN(on CelebA-HQ)	6.45	11.43	58.20	112.28	172.72	226.40

Table 2. Contrast property of GLCM calculated with all skin patches in training set.

4.1. Generalization and Robustness Analysis

Our previous experimental finding is that the trained model performs almost perfectly in *in-domain* tests. However, our further experiments show that the performance of ResNet is reduced by 22% (worst case) if the images are downsampled to 64 × 64 and JPEG compressed (Table 3: "JPEG 8x $\downarrow$"). Moreover, the model suffers more in *cross-GAN* setting, especially when the trained models are evaluated on low-resolution GANs, in which the performance dropped to around $64\% - 75\%$ (Table 4: Second row). The reduction of performance indicates that the CNN fake/real image discriminator is not robust to image editing and cannot generalize well to *cross-GAN* images, which limits its practical application.

To tackle the above problem, we further analyzed the issue. In image editing scenario, we studied the correlation between the modified images and original ones. Specifically, we calculate the Pearson Correlation Coefficient between the original image and edited images in terms of texture contrast C_d as shown in Figure 4. The coefficient value is closer to 1 as the pair distance d increases (*i.e.* larger image textures and



(a) Downsample (b) Gaussian blur (c) Gaussian noise

Figure 4. Pearson correlation coefficient of texture contrast between edited images and original images. Downsample ratio is 4, Gaussian blur kernel is 3, and Guassian noise std is 3.

more global), which indicates a strong correlation in large texture between edited and original images. In other words, large image texture has shown to be more robust to image editing. Moreover, in *cross-GAN* setting, *large* texture can also provide valuable information since the real/fake difference in terms of texture contrast still hold in the large pair distance d shown in Table 2. Thus a model that can capture long-range information is desirable to improve the model robustness and generalization ability. However, current CNN models cannot incorporate long-range information due to its small effective receptive field which is much smaller than the calculated receptive field as presented in [21].

Inspired by [6], we propose to introduce "Gram Block" into the CNN architecture and propose a novel architecture coined as Gram-Net as shown in Figure 5. The "Gram Block" captures the global texture feature and enable long-range modeling by calculating the Gram matrix in different semantic levels.

4.2. Gram-Net Architecture

The overview of Gram-Net is shown in Figure 5. Gram Blocks are added to the ResNet architecture on the input image and before every downsampling layer, incorporating global image texture information in different semantic levels. Each Gram Block contains a convolution layer to align the feature dimension from different levels, a Gram matrix calculation layer to extract global image texture feature, two conv-bn-relu layers to refine the representation, and a global-pooling layer to align the gram-style feature with ResNet backbone. The Gram matrix is calculated as follows.

$$G_{ij}^l = \sum_k F_{ik}^l F_{jk}^l \tag{2}$$

where F^l represents the l-th feature map whose spatial dimension is vectorized, and F_{ik}^l represents the kth element in the ith feature map of layer l. We show Gram matrix is a good descriptor for global or long-range texture as follows.

Can Gram matrix capture *global texture* information? In CNNs, each convolution layer l can be viewed as a filter bank, and the feature map F^l is a set of response images to these filters.

G^l is the eccentric covariance matrix of channels in layer l. Each element G_{ij}^l measures the covariance between the ith and jth vectorised response map in the layer. Equation 3 is the covariance matrix C^l of feature maps, and Gram matrix G^l in Equation 4 is actually the covariance matrix without subtracting the mean value. The diagonal elements of Gram matrix shows the response of the particular filter, while other elements show the coherence of different filters. In a word, Gram matrix is a summary of spatial statistics which discards the spatial and content information in the feature maps, and provides a stationary description of the texture.

$$C^l = (cov(F_i^l, F_j^l))_{n\times n} = (\mathbb{E}[(F_i^{lT} - \overline{F_i^{lT}})(F_j^l - \overline{F_j^l})])_{n\times n} =$$

$$\frac{1}{n-1}\begin{bmatrix} (F_1^{lT}-\overline{F_1^{lT}})(F_1^l-\overline{F_1^l}) & \cdots & (F_1^{lT}-\overline{F_1^{lT}})(F_n^l-\overline{F_n^l}) \\ \vdots & \ddots & \\ (F_n^{lT}-\overline{F_n^{lT}})(F_1^l-\overline{F_1^l}) & \cdots & (F_n^{lT}-\overline{F_n^{lT}})(F_n^l-\overline{F_n^l}) \end{bmatrix} \tag{3}$$

$$G^l = (F_i^{lT} F_j^l)_{n\times n} = \begin{bmatrix} F_1^{lT}F_1^l & \cdots & F_1^{lT}F_n^l \\ \vdots & \ddots & \\ F_n^{lT}F_1^l & \cdots & F_n^{lT}F_n^l \end{bmatrix} \tag{4}$$

In addition, G_{ij}^l is a descriptor for the whole feature map, which is not limited by the receptive field of CNNs. This property enables it to extract long-range texture feature effectively, which complements the CNN backbone.

To further analyze the information captured by Gram-Net and the CNN baseline, we adopt [22] to generate the reconstructed input that can produce the approximate feature map as the original input. The reconstructed inputs for reproducing the feature in "res-block 2" and "avg-pool" are shown

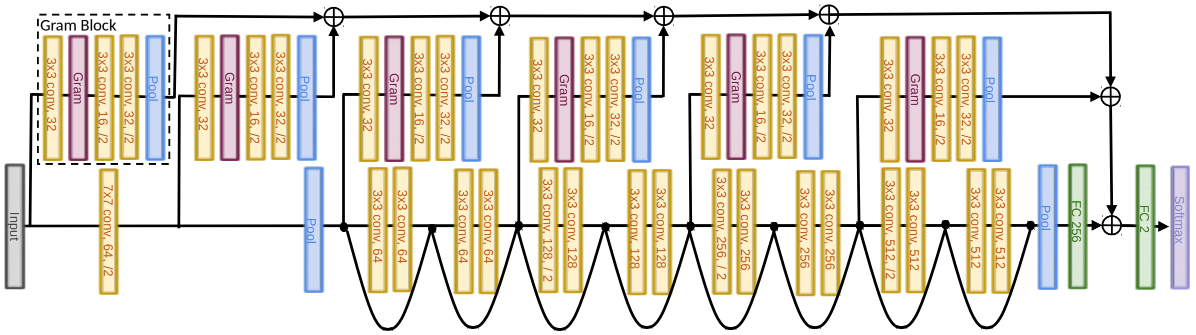
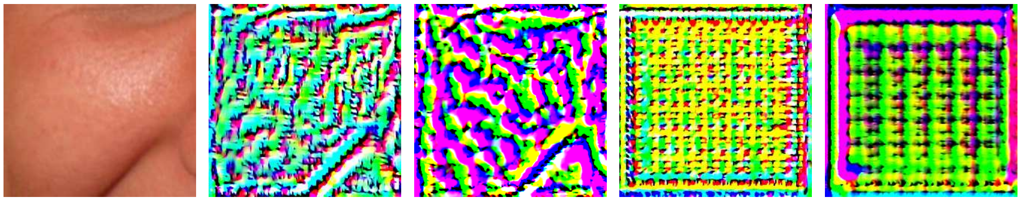


Figure 5. Gram-Net architecture. We extract global image texture feature with 6 Gram Blocks in different semantic levels from ResNet. $\oplus$ means concatenation.



(a) Original (b) ResNet (c) Gram-Net (d) ResNet (e) Gram-Net

Figure 6. Visualization of reconstructed input. Reconstructed images are multiplied by a scale factor for clearer visualization. (a) is the original image. (b)(c) are reconstructed inputs for reproducing 'res-block2' feature in ResNet and Gram-Net respectively. (d)(e) are reconstructed inputs for reproducing 'avg-pool' in ResNet and Gram-Net respectively.

in Figure 6. The texture size of the reconstructed input image from Gram-Net is larger than that of baseline ResNet, which shows that our Gram-Net captures long-range texture patterns for discrimination.

5. Experiments

Implementation details. We implement all the approaches with PyTorch [28]. Models are initialized with pretrained ImageNet weights. We train all the models with learning rate $1e^{-5}$ and select model on validation set. The validation set contains totally 800 images from DCGAN, StarGAN, CelebA, PGGAN, StyleGAN on CelebA-HQ, StyleGAN on FFHQ, CelebA-HQ and FFHQ (100 for each). In all the experiments, the models are trained on 10k real and 10k fake images and evaluated on a holdout test set containing 10k real and 10k fake images.

Experimental setup. We conduct experiments in *in-domain* and *cross-GAN* settings, and further test the models on GANs trained on other datasets (*cross-dataset*). All the images are bilinear-resized to 512×512 which serves our baseline resolution, because we found that models on this resolution already performs almost the same as 1024×1024 and can accelerate the inference. All fake images are derived by directly evaluating the author-released code and model with default parameters. We compare the performance of our Gram-Net with a recent fake face detectors Co-detect [27] and ResNet. We choose ResNet-18 as baseline because it already achieves much better performance than human beings described in Section 3.1. For a fair comparison, we implement Gram-Net and [27] with the same ResNet-18 backbone, which takes the hand-craft texture descriptor GLCM of RGB channels as input. We train these three networks with images randomly bilinear-resized into range 64×64 to 256×256 as data augmentation, and evaluate the models regarding accuracy and their robustness to image editing and cross-GAN generalization ability. To minimize the influence of randomness, we repeat each experiment five times by randomly splitting training and testing sets and show the error bar.

Robustness and cross-GAN generalization experiments on high-resolution GANs. We edit the images with downsampling and JPEG compression, which often occur unintentionally when the images are uploaded to the Internet, put into slides or used as a video frame. Specifically, the models are evaluated in the following settings. 1) Original inputs with size 512×512 ("Origin"), 2) Downsampled images to resolution 64×64 ("8x $\downarrow$"), 3) JPEG Compressed 512×512 images ("JPEG"), 4) JPEG compressed and downsampled images ("JPEG 8x $\downarrow$"). In addition, GAN and real images can be edited by adding blur or noise intentionally. In table 3, Gaussian blur ("blur") is with kernel size 25 ("blur"), and Gaussian noise ("blur") is with standard deviation 5.

The evaluation results are listed in Table 3. Our Gram-Net outperforms the compared methods in all scenarios. On aver-

Training set	Testing set	Method	Original %	8x ↓ %	JPEG %	JPEG 8x ↓	Blur %	Noise %	Avg.
StyleGAN vs. CelebA-HQ	StyleGAN vs. CelebA-HQ	Co-detect	79.93 ± 1.34	71.80 ± 1.30	74.58 ± 3.25	71.25 ±1.18	71.39 ±1.42	54.09 ± 2.45	70.51
		ResNet	96.73 ± 3.60	85.10 ± 6.22	96.68 ± 3.50	83.33 ± 5.95	79.48 ± 8.70	87.92 ± 6.16	88.20
		Gram-Net	**99.10 ± 1.36**	**95.84 ± 1.98**	**99.05 ± 1.37**	**92.39 ± 2.66**	**94.20 ± 5.57**	**92.47 ± 4.52**	**95.51**
	PGGAN vs. CelebA-HQ	Co-detect	71.22 ± 3.76	62.02 ± 2.86	64.08 ± 1.93	61.24 ± 2.28	62.46 ± 3.31	49.96 ± 0.28	61.83
		ResNet	93.74 ± 3.03	77.75 ± 4.82	89.35 ± 1.50	69.35 ± 3.25	78.06 ± 7.57	82.65 ± 2.37	81.82
		Gram-Net	**98.54** ± 1.27	**82.40 ± 6.30**	**94.65 ± 3.28**	**79.77 ± 6.13**	**91.96 ± 4.78**	**88.29 ± 3.44**	**89.26**
PGGAN vs. CelebA-HQ	PGGAN vs. CelebA-HQ	Co-detect	91.14 ± 0.61	82.94 ± 1.03	86.00 ± 1.70	82.46 ± 1.06	84.24 ± 0.93	54.77 ± 2.42	80.26
		ResNet	97.38 ± 0.52	90.87 ± 1.90	94.67 ± 1.15	89.93 ± 1.50	97.25 ± 0.87	66.60 ± 9.61	89.45
		Gram-Net	**98.78 ± 0.49**	**94.66 ± 3.10**	**97.29 ± 1.05**	**94.08 ± 3.22**	**98.55 ± 0.92**	**70.32 ± 12.04**	**92.28**
	StyleGAN vs. CelebA-HQ	Co-detect	57.30 ± 1.62	57.41 ± 0.85	52.90 ± 1.67	82.46 ± 1.06	57.41 ± 0.93	50.08 ± 0.10	51.47
		ResNet	97.98 ± 1.90	87.91 ± 1.01	92.03 ± 4.14	82.23 ± 1.39	94.79 ± 1.32	**60.89 ± 7.24**	85.97
		Gram-Net	**98.55 ± 0.89**	**91.57 ± 2.95**	**94.28 ± 3.67**	**83.64 ± 3.43**	**97.05 ± 1.04**	60.07 ± 7.32	**87.52**
StyleGAN vs. FFHQ	StyleGAN vs. FFHQ	Co-detect	69.73 ± 2.41	67.27 ± 1.68	67.48 ± 2.83	64.65 ± 1.67	64.55 ± 1.93	54.66 ± 3.97	64.74
		ResNet	90.27 ± 3.05	70.99 ± 1.13	89.35 ± 3.42	67.96 ± 1.13	**75.60 ± 10.75**	81.32 ± 5.06	81.50
		Gram-Net	**98.96 ± 0.51**	**89.22 ± 4.44**	**98.69 ± 0.81**	**87.86 ± 3.42**	70.99 ± 6.07	**94.27 ± 2.12**	**90.00**

Table 3. Performance on in-domain and cross to high-resolution GANs. In each training setting, the first half shows results in the *in-domain* setting and the second half shows results in the *cross-GAN* setting. Column (Avg.) shows the averaged results across all settings. The accuracy in "Original %" column is lower than the results in Table 1 because the models are selected to achieve best average performance in all the settings with validation set. The average inference time for one image on RTX 2080 Ti are as follows. Gram-Net takes $2.40e^{-3}$s, ResNet-18 takes $2.35e^{-3}$s, and Co-detect [27] takes $8.68e^{-3}$s, in which $6.57e^{-3}$s for co-occurance matrix calculation.

age, it outperforms [27] by more than 20%. The results show that our Gram-Net adaptively extracts robust texture representation in feature space, which is much more powerful than low-level texture representations such as GLCM. Our model also improves the ResNet baseline by around 7% (on average) in both in-domain and cross-GAN settings trained on StyleGAN vs. CelebA-HQ. The reason why Gram-Net improves less when trained on PGGAN vs. CelebA-HQ can be partially explained according to the GLCM statistics shown in Table 2. Images generated by PGGAN have larger $\mathcal{C}_d$ than StyleGAN, which is closer to real images.

The above results manifest the effectiveness of Gram-Net in extracting features more invariant to different GAN models and more robust to image editing operations, such as downsampling, JPEG compression, blur and noise.

Generalize to low-resolution GANs. To further evaluate the models' generalization capability, we directly apply the models above to low-resolution GANs trained on CelebA. We randomly choose 10k images from each set to evaluate our model. The fake images are kept at their original sizes, *i.e.*, 64×64 for DCGAN and DRAGAN, 128×128 for StarGAN. CelebA images are of size 178×218, so we center crop the 178×178 patch in the middle to make it square.

The results as listed in Table 4 show that our Gram-Net better generalizes to low-resolution GANs. The performance of baseline ResNet and [27] degrades to around 50% to 75% in this setting. However, our method outperforms the ResNet baseline by around 10% and [27] by around 15% regarding accuracy in all settings. This further demonstrates global image texture feature introduced by our "Gram Block" is more invariant across different GANs, which can even generalize to detect fake faces from image-to-image translation model – StarGAN.

Method	Accuracy
Co-detect	59.81 ± 10.82
ResNet	80.55 ± 6.37
Gram-Net	**93.35 ± 2.25**

Table 5. Performance of Gram-Net when StyleGAN discriminator contains Gram-Block. The models are trained on StyleGAN (origin) vs. CelebA-HQ and tested on StyleGAN (with Gram-Block in discriminator) vs. CelebA-HQ.

Generalize to StyleGAN trained with Gram-Block in discriminator. In this section, we evaluate the model on images from GAN models whose discriminator also contains Gram Blocks. We fine-tune StyleGAN with extra Gram-Blocks inserted in the discriminator, and further evaluate whether Gram-Net still works in this setting. We add 8 identical Gram-Blocks as in Gram-Net to encode feature maps (from feature map size 1024 to 4) in StyleGAN discriminator, and concatenate the 8×32 dimension feature vector extracted by Gram-Blocks with the original 512 dimension feature vector in original discriminator before the final classification. We fine-tune the model for 8K epochs on CelebA-HQ initialized by the author released model. We evaluate 10K fake images from StyleGAN with Gram-Block in discriminator and 10K images from CelebA-HQ. The images are resized to 512 × 512 resolution. We directly apply the models used in Table 3 and 4 in this setting.

The results in Table 5 show that our Gram-Net still outperforms baseline methods even though the Gram-Block is inserted in the GAN discriminator. This demonstrates that our findings and analysis in section 3.3 are still valid.

Test Train	Method	DCGAN vs. CelebA %	DRAGAN vs. CelebA %	StarGAN vs. CelebA %	Avg.
StyleGAN	Co-detect	68.83 ± 9.57	59.99 ± 8.81	58.60 ± 3.99	62.47
vs.	ResNet	75.11 ± 8.10	65.53 ± 8.20	64.04 ± 7.69	68.22
CelebA-HQ	Gram-Net	**81.65 ± 3.51**	**76.40 ± 6.06**	**74.96 ± 4.90**	**77.67**

Table 4. Performance of Gram-Net on generalization to low-resolution GANs.

Method	Train on StyleGAN vs. CelebA-HQ Test on StyleGAN vs. FFHQ	Train on PGGAN vs. CelebA-HQ Test on StyleGAN vs. FFHQ	Train on StyleGAN vs. FFHQ Test on StyleGAN vs. CelebA-HQ
Co-detect	48.90 ± 3.95	48.71 ± 1.43	59.22 ± 1.30
ResNet	75.45 ± 7.01	54.44 ±3.64	80.14 ± 7.47
Gram-Net	**77.69 ± 6.49**	**59.57 ± 8.07**	**80.72 ± 6.02**

Table 6. Performance of Gram-Net in cross-dataset settings

distance	1	2	5	10	15	20
ImageNet	**525.70**	**676.60**	**1551.85**	**2267.16**	**2892.90**	**3334.14**
BigGAN	367.65	536.81	1426.66	2146.90	2771.96	3207.97

Table 7. Contrast property of GLCM calculated with BigGAN and ImageNet images in training set with different pair distances.

Cross-dataset experiments. Cross-dataset generalization is a challenging problem due to the inherent difference in dataset construction. Our experiments show that the statistics of CelebA-HQ and FFHQ are significantly different and can easily be distinguished by a neural network. Specifically, we built a real face image dataset consisting of 10K CelebA-HQ images and 10K FFHQ images, and our further experiments show that a ResNet network can achieve more than 99.9% accuracy to discriminate CelebA-HQ and FFHQ images. This experiment shows that real face datasets significantly differ from each other.

Despite the fact above, we evaluate our Gram-Net and baseline approaches in the cross-dataset setting as follows: train on StyleGAN(PGGAN) vs. CelebA-HQ and test on StyleGAN vs. FFHQ, train on StyleGAN vs. FFHQ and test on StyleGAN vs. CelebA-HQ. We keep all of the images as their original resolution in this experiment. The models are the same with the ones in Table 3 and 4.

The result in Table 6 shows that fake image detectors trained on more realistic dataset (FFHQ) and stronger GANs (StyleGAN) have stronger ability to cross to less realistic datasets (CelebA-HQ) and less strong GANs (PGGAN). Also, Gram-Net still outperforms baselines methods.

Generalize to natural images. In this section, we extend our analysis and apply Gram-Net to fake/real natural images. Specifically, we analyze ImageNet [16] *vs.* BigGAN [3], where the BigGAN model is trained on ImageNet.

To analyze fake/real natural images, we further employ GLCM. We find that the difference in terms of texture contract between fake and real face images also holds for natural images. As Table 7 shows, real images retain stronger texture contrast than GAN images for all the distances measured.

To evaluate the generalization ability of our model trained on face images, we directly apply the model used in Table 3 and 4 to test 10K ImageNet and 10K BigGAN images (10 images each class), and the results are shown in Table 8.

Training set	Testing set	Method	Accuracy % .
StyleGAN	ImageNet	Co-detect [27]	51.94 ± 2.31
vs.	*vs.*	ResNet	71.93 ± 2.09
CelebA-HQ	BigGAN	Gram-Net	**80.29 ± 3.20**

Table 8. Quantitative results on ImageNet vs BigGAN.

6. Conclusion

In this paper, we conduct empirical studies on human and CNNs in discriminating fake/real faces and find that fake faces attain different textures from the real ones. Then, we perform low-level texture statistical analysis to further verify our findings. The statistics also show that *large texture* information is more robust to image editing and invariant among different GANs. Motivated by these findings, we propose a new architecture – Gram-Net, which leverages *global texture* features to improve the robustness and generalization ability in fake face detection. Experimental results show that Gram-Net significantly outperforms the most recent approaches and baseline models in all settings including *in-domain*, *cross-GAN*, and *cross-dataset*. Moreover, our model exhibits better generalization ability in detecting fake natural images. Our work shows a new and promising direction for understanding fake images from GANs and improving fake face detection in the real world.

7. Acknowledgement

This work was supported by the ERC grant ERC-2012-AdG 321162-HELIOS, EPSRC grant Seebibyte EP/M013774/1 and EPSRC/MURI grant EP/N019474/1. We would also like to acknowledge the Royal Academy of Engineering and FiveAI.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223, 2017.

[2] David Berthelot, Thomas Schumm, and Luke Metz. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017.

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, 2018.

[5] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in neural information processing systems*, pages 262–270, 2015.

[6] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

[7] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[9] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017.

[10] Robert M Haralick, Karthikeyan Shanmugam, et al. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[13] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948*, 2018.

[14] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[15] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira. On convergence and stability of gans. *arXiv preprint arXiv:1705.07215*, 2017.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[17] Haodong Li, Bin Li, Shunquan Tan, and Jiwu Huang. Detection of deep network generated images using disparities in color components. *arXiv preprint arXiv:1808.07276*, 2018.

[18] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts. *arXiv preprint arXiv:1811.00656*, 2018.

[19] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems*, pages 700–708, 2017.

[20] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

[21] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *Advances in neural information processing systems*, pages 4898–4906, 2016.

[22] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015.

[23] Francesco Marra, Diego Gragnaniello, Davide Cozzolino, and Luisa Verdoliva. Detection of gan-generated fake images over social networks. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 384–389. IEEE, 2018.

[24] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do gans leave artificial fingerprints? In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 506–511. IEEE, 2019.

[25] Francesco Marra, Cristiano Saltori, Giulia Boato, and Luisa Verdoliva. Incremental learning for the detection and classification of gan-generated images. *arXiv preprint arXiv:1910.01568*, 2019.

[26] Scott McCloskey and Michael Albright. Detecting gan-generated imagery using color cues. *arXiv preprint arXiv:1812.08247*, 2018.

[27] Lakshmanan Nataraj, Tajuddin Manhar Mohammed, BS Manjunath, Shivkumar Chandrasekaran, Arjuna Flenner, Jawadul H Bappy, and Amit K Roy-Chowdhury. Detecting gan generated fake images using co-occurrence matrices. *arXiv preprint arXiv:1903.06836*, 2019.

[28] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[29] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[30] Run Wang, Lei Ma, Felix Juefei-Xu, Xiaofei Xie, Jian Wang, and Yang Liu. Fakespotter: A simple baseline for spotting ai-synthesized fake faces. *arXiv preprint arXiv:1909.06122*, 2019.

[31] Li Xu, Cewu Lu, Yi Xu, and Jiaya Jia. Image smoothing via l 0 gradient minimization. In *ACM Transactions on Graphics (TOG)*, volume 30, page 174. ACM, 2011.

[32] Xinsheng Xuan, Bo Peng, Jing Dong, and Wei Wang. On the generalization of gan image forensics. *arXiv preprint arXiv:1902.11153*, 2019.

[33] Xin Yang, Yuezun Li, Honggang Qi, and Siwei Lyu. Exposing gan-synthesized faces using landmark locations. *arXiv preprint arXiv:1904.00167*, 2019.

[34] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. *arXiv preprint arXiv:1907.06515*, 2019.

[35] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[36] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.